AMENDMENTS TO THE SPECIFICATION

Please amend the claims as follows.

1. (Currently Amended) A method for accuracy-aware analysis of a program, comprising:

obtaining source code for the program comprising a floating-point variable;

instrumenting the source code to associate an accuracy-aware tracking structure with the floating-point variable to obtain instrumented source code with functionality to call a runtime logging utility;

compiling the [[to]] instrumented source code to obtain instrumented compiled code; [[and]]

executing the instrumented compiled code, wherein executing the instrumented compiled code comprises executing one of a plurality of operations on the floating-point variable to obtain a resultant value for the floating-point variable; and ~~using the accuracy-aware tracking structure to track a plurality of operations applied to the floating-point variable,~~

executing the runtime logging utility to populate the accuracy-aware tracking structure, wherein for each operation performed on the floating-point variable, the runtime logging utility is configured to:

~~wherein data stored in the accuracy-aware tracking structure is used to determine error resulting from applying the plurality of operations and which of the plurality of operations caused the error.~~

increment an operations variable in the accuracy-aware tracking structure corresponding to the operation performed on the floating-point variable,

determine a scaled mantissa for the resultant value,

compare the scaled mantissa with the resultant value to determine whether the resultant value is exact,

quantify the error associated with the resultant value when the resultant value is not exact to obtain an error value associated with the resultant value,

store the resultant value, the scaled mantissa, and the error value when the resultant value is not exact, and

store the resultant value and the scaled mantissa when the resultant value is exact.

2. (Original) The method of claim 1, further comprising:

generating an accuracy-aware analysis report using the accuracy-aware tracking structure.

3. (Currently Amended) The method of claim 2, wherein the accuracy-aware analysis report includes at least one ~~tracking variable associated with the floating-point variable~~ selected from the group consisting of [[an]] the error value ~~variable~~, the [[a]] scaled mantissa ~~digits variable~~, and the resultant value ~~a renormalization variable, a left digit destruction variable, and an operations variable~~.

4. (Currently Amended) The method of claim [[3]] 1, wherein the error ~~variable~~ value is a half unit in last place (HULP) value associated with the floating-point variable, wherein the HULP value is a base of a floating-point representation raised to a power of the number of bits causing the error in the floating-point variable.

5. (Previously Presented) The method of claim 4, wherein the HULP value is determined using information obtained during renormalization.

6. (Currently Amended) The method of claim 3, wherein the error ~~variable~~ value comprises an upper limit interval variable or a lower limit interval variable.

7. (Currently Amended) The method of claim [[3]] 1, wherein [[an]] the operations variable comprises at least one selected from the group consisting of a multiplication variable, a division variable, and a square root variable.

8. (Currently Amended) The method of claim [[3]] 1, wherein the accuracy-aware tracking structure further comprises a [[the]] renormalization variable which track[[s]] the number of addition and subtraction operations performed on the floating-point variable that do not involve left digit destruction.

9. (Cancelled)

10. (Currently Amended) The method of claim 1 [[9]], further comprising:

updating the error ~~variable~~ value using data obtained from quantifying the error associated with the resultant value, if the resultant value is not exact.

11. (Currently Amended) The method of claim 1 [[9]], further comprising:

determining whether the resultant value exceeds an accuracy threshold if the resultant value is not exact.

12. (Original) The method of claim 11, wherein execution of the compiled instrumented code halts if the accuracy threshold hold is exceeded.

13. (Original) The method of claim 11, wherein the accuracy threshold comprises at least one selected from the group consisting of a relative error threshold, an absolute error threshold, and a comparison test.

14. (Original) The method of claim 1, further comprising:

setting an accuracy threshold for the program.

15. (Currently Amended) The method of claim 1, wherein instrumenting the source code comprises:

parsing the source code to obtain the floating-point variable; and

inserting additional source code to call the runtime logging utility. ~~update the accuracy-aware tracking structure associated with the floating-point variable.~~

16. (Cancelled)

17. (Original) The method of claim 1, wherein the floating-point variable is double type.

18. – 29. (Cancelled)

30. (Currently Amended) A computer system for performing accuracy-aware analysis on a program, comprising:

a processor;

a memory;

a storage device; and

software instructions stored in the memory for enabling the computer system under control of the processor, to:

obtain source code for the program comprising a floating-point variable;

instrument the source code to associate an accuracy-aware tracking structure with the floating-point variable to obtain instrumented source code with functionality to call a runtime logging utility;

compile the [[to]] instrumented source code to obtain instrumented compiled code; [[and]]

execute the instrumented compiled code, wherein executing the instrumented compiled code comprises executing one of a plurality of operations on the floating-point variable to obtain a resultant value for the floating-point variable; and ~~using the accuracy-aware tracking structure to track a plurality of operations applied to the floating-point variable,~~

execute the runtime logging utility to populate the accuracy-aware tracking structure wherein for each operation performed on the floating-point variable, the runtime logging utility is configured to:

~~wherein data stored in the accuracy-aware tracking structure is used to determine error resulting from applying the plurality of operations and which of the plurality of operations caused the error.~~

increment an operations variable in the accuracy-aware tracking structure corresponding to the operation performed on the floating-point variable,

determine a scaled mantissa for the resultant value,

compare the scaled mantissa with the resultant value to determine whether the resultant value is exact,

quantify the error associated with the resultant value when the resultant value is not exact to obtain an error value associated with the resultant value,

store the resultant value, the scaled mantissa, and the error value when the resultant value is not exact, and

store the resultant value and the scaled mantissa when the resultant value is exact.

31. (Previously Presented) The computer system of claim 30, further comprising software instructions to:

generate an accuracy-aware analysis report using the accuracy-aware tracking structure.

32. (Currently Amended) The computer system of claim 31, wherein the accuracy-aware analysis report includes at least one ~~tracking variable associated with the floating-point variable~~ selected from the group consisting of [[an]] the error value ~~variable~~, the [[a]] scaled mantissa ~~digits variable~~, and the resultant value ~~a renormalization variable, a left digit destruction variable, and an operations variable~~.

33. (Currently Amended) The computer system of claim [[32]] 30, wherein the error ~~variable~~ value is a half unit in last place (HULP) value associated with the floating-point variable, wherein the HULP value is a base of a floating-point representation raised to a power of the number of bits causing the error in the floating-point variable.

34. (Previously Presented) The computer system of claim 33, wherein the HULP value is determined using information obtained during renormalization.

35. (Currently Amended) The computer system of claim [[32]] 30, wherein [[an]] the operations variable comprises at least one selected from the group consisting of a multiplication variable, a division variable, and a square root variable.

36. (Cancelled)

37. (Currently Amended) The computer system of claim 30, wherein software instructions for instrumenting the source code comprises software instructions to:

parse the source code to obtain the floating-point variable; and

insert additional source code to call the runtime logging utility. ~~update the accuracy-aware tracking structure associated with the floating-point variable.~~